Polynomial Systems Theories in Biology
Talk at ACA 2022

James Davenport[1]

University of Bath and DEWCAD group
https://matthewengland.coventry.domains/dewcad/index.html
EPSRC Grants EP/T015748/1 and EP/T015713/1

[1]Partially supported by EU H2020 project SC^2 (712689), and wholly supported by DEWCAD colleagues

Plan of Talk

1. Background
2. Parametric Occurrence of Multiple Steady States [BDE+17, EEG+17, BDE+19]
3. Expected number of positive real solutions in reaction networks [FS20]
4. Personal Conclusions

Prehistory at Bath

Prehistory at Bath

1988 "*Solution of Some Equations in Biochemistry*" [BDS88] — rejected by *J. Theoretical Biology* as "this is too theoretical".

Prehistory at Bath

1988 "*Solution of Some Equations in Biochemistry*" [BDS88] — rejected by *J. Theoretical Biology* as "this is too theoretical".

1991 "*Computer Algebra Approaches to Enzyme Kinetics*" [BDD+91] — let's pretend it's Control Theory.

Prehistory at Bath

1988 "*Solution of Some Equations in Biochemistry*" [BDS88] — rejected by *J. Theoretical Biology* as "this is too theoretical".

1991 "*Computer Algebra Approaches to Enzyme Kinetics*" [BDD+91] — let's pretend it's Control Theory.

1993 "*Solution of Some Equations in Biochemistry*" Mustafa Bayram's thesis —[Bay93].

Prehistory at Bath

1988 "*Solution of Some Equations in Biochemistry*" [BDS88] — rejected by *J. Theoretical Biology* as "this is too theoretical".

1991 "*Computer Algebra Approaches to Enzyme Kinetics*" [BDD+91] — let's pretend it's Control Theory.

1993 "*Solution of Some Equations in Biochemistry*" Mustafa Bayram's thesis —[Bay93].

Prehistory at Bath

1988 "*Solution of Some Equations in Biochemistry*" [BDS88] — rejected by *J. Theoretical Biology* as "this is too theoretical".

1991 "*Computer Algebra Approaches to Enzyme Kinetics*" [BDD+91] — let's pretend it's Control Theory.

1993 "*Solution of Some Equations in Biochemistry*" Mustafa Bayram's thesis —[Bay93].

We could show that there was scope for applying computer algebra to enzyme kinetic reactions.

Polynomial Systems Theories

Why do I say "theories"?

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]
- $*$ Can also be computed via Regular Chains [CM14]

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]
- $*$ Can also be computed via Regular Chains [CM14]

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]
- $*$ Can also be computed via Regular Chains [CM14]

Only $\mathbf{C}_1$ was available in easy software at the time of our early work (and it is still the most accessible — in all computer algebra systems).

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]
- $*$ Can also be computed via Regular Chains [CM14]

Only $\mathbf{C}_1$ was available in easy software at the time of our early work (and it is still the most accessible — in all computer algebra systems).

None of these quite meet need Biology's needs, where almost all variables (concentrations, populations etc.) are in $\mathbf{R}_{\geq 0}$.

Polynomial Systems Theories

Why do I say "theories"?

- $\mathbf{C}_1$ Equations from $\mathbf{Q}[x_1, \ldots, x_n]$, solutions in $\mathbf{C}$ — Gröbner Bases [Buc65, CLO15]
- $\mathbf{C}_2$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{C}$ — Regular Chains [Wu 78, ALM99]
- $\mathbf{R}$ Equations from $\mathbf{Q}[x_1] \ldots [x_n]$, solutions in $\mathbf{R}$ — Cylindrical Algebraic Decomposition [Col75]
- $*$ Can also be computed via Regular Chains [CM14]

Only $\mathbf{C}_1$ was available in easy software at the time of our early work (and it is still the most accessible — in all computer algebra systems).
None of these quite meet need Biology's needs, where almost all variables (concentrations, populations etc.) are in $\mathbf{R}_{\geq 0}$.
All are doubly-exponential worst case in the number of variables (including parameters): [BD07, DH88, MM82, MR13].

Parametric Occurrence of Multiple Steady States [BDE+17, EEG+17, BDE+19]

- We aim to identify symbolically regions of a parameter space over which a biological network exhibits multi-stationarity (multiple steady states).

[2] `http://www.ebi.ac.uk/biomodels-main/`

Parametric Occurrence of Multiple Steady States [BDE+17, EEG+17, BDE+19]

- We aim to identify symbolically regions of a parameter space over which a biological network exhibits multi-stationarity (multiple steady states).
- When the corresponding reactions are modelled by mass action kinetics, then mathematically the task is to (a) identify positive real solutions of a parametrised system of polynomials and (b) check stability. We focus on task (a).

[2] `http://www.ebi.ac.uk/biomodels-main/`

Parametric Occurrence of Multiple Steady States [BDE+17, EEG+17, BDE+19]

- We aim to identify symbolically regions of a parameter space over which a biological network exhibits multi-stationarity (multiple steady states).
- When the corresponding reactions are modelled by mass action kinetics, then mathematically the task is to (a) identify positive real solutions of a parametrised system of polynomials and (b) check stability. We focus on task (a).
- Specifically, we consider the Mitogen-Activated Protein Kinases (MAPK) cascade. We have results for models # 26 (and # 28) in the Biomodels Database[2].

[2] `http://www.ebi.ac.uk/biomodels-main/`

Parametric Occurrence of Multiple Steady States [BDE+17, EEG+17, BDE+19]

- We aim to identify symbolically regions of a parameter space over which a biological network exhibits multi-stationarity (multiple steady states).
- When the corresponding reactions are modelled by mass action kinetics, then mathematically the task is to (a) identify positive real solutions of a parametrised system of polynomials and (b) check stability. We focus on task (a).
- Specifically, we consider the Mitogen-Activated Protein Kinases (MAPK) cascade. We have results for models # 26 (and # 28) in the Biomodels Database[2].
- In contrast to most of the literature on the topic, we work with methods from Symbolic Computation (where values are exact rather than floating point).

[2] `http://www.ebi.ac.uk/biomodels-main/`

MAPK – what and why?

A Mitogen-Activated Protein Kinase (MAPK) is a type of protein kinase enzyme.

MAPK – what and why?

A Mitogen-Activated Protein Kinase (MAPK) is a type of protein kinase enzyme. Why study MAPK?

MAPK – what and why?

A Mitogen-Activated Protein Kinase (MAPK) is a type of protein kinase enzyme. Why study MAPK?

- MAPKs are involved in directing cellular responses to a diverse array of stimuli, such as mitogens, osmotic stress and heat shock.
- They regulate cell functions including proliferation, gene expression, differentiation and mitosis.

MAPK – what and why?

A Mitogen-Activated Protein Kinase (MAPK) is a type of protein kinase enzyme. Why study MAPK?

- MAPKs are involved in directing cellular responses to a diverse array of stimuli, such as mitogens, osmotic stress and heat shock.
- They regulate cell functions including proliferation, gene expression, differentiation and mitosis.

Why study multistationarity?

MAPK – what and why?

A Mitogen-Activated Protein Kinase (MAPK) is a type of protein kinase enzyme. Why study MAPK?

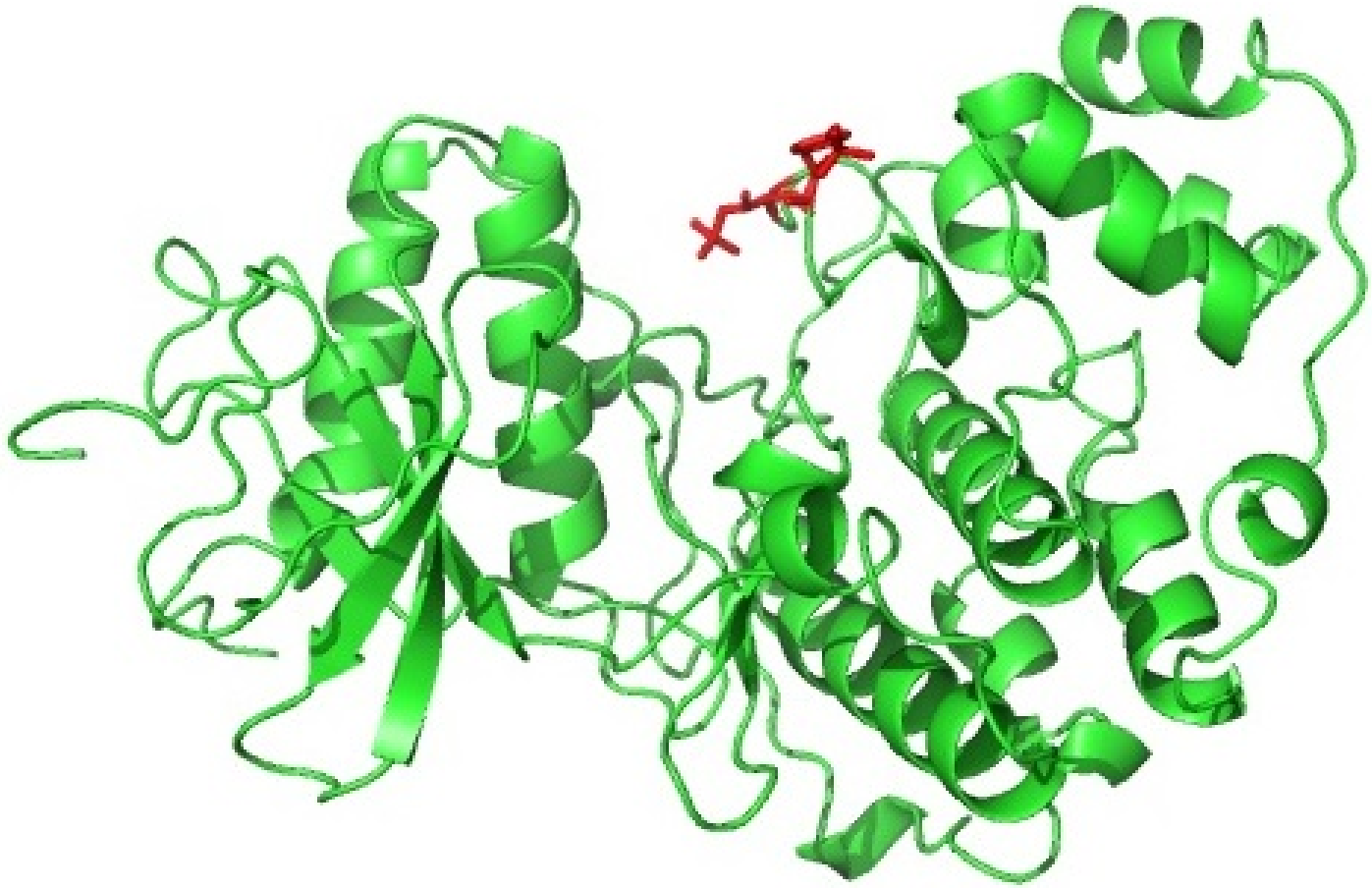
- MAPKs are involved in directing cellular responses to a diverse array of stimuli, such as mitogens, osmotic stress and heat shock.
- They regulate cell functions including proliferation, gene expression, differentiation and mitosis.

Why study multistationarity?

- Instrumental to cellular memory and cell differentiation during development or regeneration of multicellular organisms.
- Used by micro organisms in survival strategies.

MAP: Phosphorylated residues are displayed in red

X-ray structure of the ERK2 MAP kinase in its active form



Source: Wikipedia - via molecular visualization system `PyMol`.

This is work of an interdisciplinary group including researchers from Mathematics, Computer Science, and Systems Biology. Naturally, our focus here will be on symbolic computation aspects.

Why use symbolic methods for this problem?

- Numerical methods observed to give incorrect results at certain points in parameter space.
- Symbolic methods have the scope to give semi-algebraic descriptions of parameter space: the exact solution.

Case Study: Model 26

From: `www.ebi.ac.uk/biomodels-main/BIOMD0000000026`

$$\dot{x}_1 = k_2x_6 + k_{15}x_{11} - k_1x_1x_4 - k_{16}x_1x_5$$
$$\dot{x}_2 = k_3x_6 + k_5x_7 + k_{10}x_9 + k_{13}x_{10} - x_2x_5(k_{11} + k_{12}) - k_4x_2x_4$$
$$\dot{x}_3 = k_6x_7 + k_8x_8 - k_7x_3x_5$$
$$\dot{x}_4 = x_6(k_2 + k_3) + x_7(k_5 + k_6) - k_1x_1x_4 - k_4x_2x_4$$
$$\dot{x}_5 = k_8x_8 + k_{10}x_9 + k_{13}x_{10} + k_{15}x_{11} - x_2x_5(k_{11} + k_{12}) - k_7x_3x_5 - k_{16}x_1x_5$$
$$\dot{x}_6 = k_1x_1x_4 - x_6(k_2 + k_3)$$
$$\dot{x}_7 = k_4x_2x_4 - x_7(k_5 + k_6)$$
$$\dot{x}_8 = k_7x_3x_5 - x_8(k_8 + k_9)$$
$$\dot{x}_9 = k_9x_8 - k_{10}x_9 + k_{11}x_2x_5$$
$$\dot{x}_{10} = k_{12}x_2x_5 - x_{10}(k_{13} + k_{14})$$
$$\dot{x}_{11} = k_{14}x_{10} - k_{15}x_{11} + k_{16}x_1x_5$$

11 differential equations

11 variables

16 parameters

Rate Constants

The biomodels database also gives us meaningful values for the rate constants.

- Some are measured:

$$k_1 = 0.02, \quad k_3 = 0.01, \quad k_4 = 0.032,$$
$$k_7 = 0.045, \quad k_9 = 0.092, \quad k_{11} = 0.01,$$
$$k_{12} = 0.01, \quad k_{15} = 0.086, \quad k_{16} = 0.0011.$$

- Others are estimated with confidence:

$$k_2 = 1, \quad k_5 = 1, \quad k_6 = 15, \quad k_8 = 1,$$
$$k_{10} = 1, \quad k_{13} = 1, \quad k_{14} = 0.5.$$

Three further Linear Conservation Constraints may be derived, introducing three further constant parameters.

$$\begin{aligned} x_5 + x_8 + x_9 + x_{10} + x_{11} &= k_{17} \\ x_4 + x_6 + x_7 &= k_{18} \\ x_1 + x_2 + x_3 + x_6 + x_7 + x_8 + x_9 + x_{10} + x_{11} &= k_{19} \end{aligned}$$

We work with some realistic values for these new parameters:

$$k_{17} = 100, \qquad k_{18} = 50, \qquad k_{19} \in \{200, 500\}.$$

However, the confidence in these estimates is not as high as the others. Ideally we would treat all three of these as symbolic parameters.

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;
- Substituting for values of parameters:

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;
- Substituting for values of parameters:
  - Ideally all but k_{17}, k_{18}, k_{19};

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;
- Substituting for values of parameters:
  - Ideally all but k_{17}, k_{18}, k_{19};
  - In [BDE+17] it was all but one of these.

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;
- Substituting for values of parameters:
  - Ideally all but k_{17}, k_{18}, k_{19};
  - In [BDE+17] it was all but one of these.
- Converting to rationals and multiplying up to integers.

Semi-algebraic System of Interest I

To identify regions of multistationarity we must count real (ideally positive) solutions of an integer polynomial system:

- Replacing the left hand sides of Model 26 by 0;
- Supplementing with the linear conservation constraints;
- Substituting for values of parameters:
  - Ideally all but k_{17}, k_{18}, k_{19};
  - In [BDE+17] it was all but one of these.
- Converting to rationals and multiplying up to integers.
- Appending positivity constraints on all variables and free parameters.

Semi-algebraic System of Interest II

$$0 = -200x_1x_4 - 11x_1x_5 + 860x_{11} + 10000x_6,$$
$$0 = -16x_2x_4 - 10x_2x_5 + 500x_{10} + 5x_6 + 500x_7 + 500x_9,$$
$$0 = -9x_3x_5 + 3000x_7 + 200x_8,$$
$$0 = -10x_1x_4 - 16x_2x_4 + 505x_6 + 8000x_7,$$
$$0 = -11x_1x_5 - 200x_2x_5 - 450x_3x_5 + 10000x_{10} + 860x_{11} + 10000x_8 + 10000x_9,$$
$$0 = 2x_1x_4 - 101x_6,$$
$$0 = 4x_2x_4 - 2000x_7,$$
$$0 = 45x_3x_5 - 1092x_8,$$
$$0 = 5x_2x_5 + 46x_8 - 500x_9,$$
$$0 = x_2x_5 - 150x_{10},$$
$$0 = 11x_1x_5 + 5000x_{10} - 860x_{11},$$
$$0 = -k_{17} + x_{10} + x_{11} + x_5 + x_8 + x_9,$$
$$0 = -k_{18} + x_4 + x_6 + x_7,$$
$$0 = -k_{19} + x_1 + x_{10} + x_{11} + x_2 + x_3 + x_6 + x_7 + x_8 + x_9,$$
$$0 < x_1, \ldots, 0 < x_{11}, 0 < k_{17}, 0 < k_{18}, 0 < k_{19}.$$

14 polynomial equations
11 variables
$1 - 3$ parameters
$12 - 14$ positivity conditions
denote (conjunction of) this as φ

What symbolic methods do we use?

Tools designed for studying real solutions of polynomial **systems** (i.e. including inequalities and inequations - not just ideals).

What symbolic methods do we use?

Tools designed for studying real solutions of polynomial **systems** (i.e. including inequalities and inequations - not just ideals).

- Cylindrical Algebraic Decomposition (CAD). Developed by Collins [Col75] and his students starting in the 1970s, and heavily developed since. Numerous implementations: Mathematica, ProjectionCAD, Qepcad-B, Redlog, RegularChains, SyNRAC, Maple.

What symbolic methods do we use?

Tools designed for studying real solutions of polynomial **systems** (i.e. including inequalities and inequations - not just ideals).

- Cylindrical Algebraic Decomposition (CAD). Developed by Collins [Col75] and his students starting in the 1970s, and heavily developed since. Numerous implementations: Mathematica, ProjectionCAD, Qepcad-B, Redlog, RegularChains, SyNRAC, Maple.
- Virtual Substitution (VS). Developed by Weispfenning [Wei88, Wei94] and his students starting in the late 1980s. Leading implementation in Redlog [DSS04] and Maple [Ton21].

What symbolic methods do we use?

Tools designed for studying real solutions of polynomial **systems** (i.e. including inequalities and inequations - not just ideals).

- Cylindrical Algebraic Decomposition (CAD). Developed by Collins [Col75] and his students starting in the 1970s, and heavily developed since. Numerous implementations: Mathematica, ProjectionCAD, Qepcad-B, Redlog, RegularChains, SyNRAC, Maple.
- Virtual Substitution (VS). Developed by Weispfenning [Wei88, Wei94] and his students starting in the late 1980s. Leading implementation in Redlog [DSS04] and Maple [Ton21].
- Lazy Real Triangularize (LRT). Recent work by Chen *et al.* [CDM+11, CM16]. Implemented in the RegularChains Library for Maple.

A CAD is:

- a decomposition meaning a partition of $\mathbf{R}^n$ into connected subsets called cells;
- (semi)-algebraic meaning that each cell can be defined by a sequence of polynomial equations and inequalities.
- cylindrical meaning the cells are arranged in a useful manner - their projections (relative to a given variable ordering) are either equal or disjoint.

Produced from a set of polynomials so each has constant sign $(+/0/-)$ in each cell (thus truth of overall system also constant).

A CAD is:

- a decomposition meaning a partition of $\mathbf{R}^n$ into connected subsets called cells;
- (semi)-algebraic meaning that each cell can be defined by a sequence of polynomial equations and inequalities.
- cylindrical meaning the cells are arranged in a useful manner - their projections (relative to a given variable ordering) are either equal or disjoint.

Produced from a set of polynomials so each has constant sign $(+/0/-)$ in each cell (thus truth of overall system also constant).

CAD is necessary, and theoretically sufficient to solve the problem, but used alone is computationally infeasible. We found success when combining with either VS or LRT (focus on latter here).

A CAD is:

- a decomposition meaning a partition of $\mathbf{R}^n$ into connected subsets called cells;
- (semi)-algebraic meaning that each cell can be defined by a sequence of polynomial equations and inequalities.
- cylindrical meaning the cells are arranged in a useful manner - their projections (relative to a given variable ordering) are either equal or disjoint.

Produced from a set of polynomials so each has constant sign $(+/0/-)$ in each cell (thus truth of overall system also constant).

A CAD is:

- a decomposition meaning a partition of $\mathbf{R}^n$ into connected subsets called cells;
- (semi)-algebraic meaning that each cell can be defined by a sequence of polynomial equations and inequalities.
- cylindrical meaning the cells are arranged in a useful manner - their projections (relative to a given variable ordering) are either equal or disjoint.

Produced from a set of polynomials so each has constant sign $(+/0/-)$ in each cell (thus truth of overall system also constant).

CAD is necessary, and theoretically sufficient to solve the problem, but used alone is computationally infeasible. We found success when combining with either VS or LRT (focus on latter here).

Approach 1: CAD + VS in Redlog

Consider $\varphi_{k_{19}}$ as the system with all parameters except k_{19} set.

- We solve for $i \in \{1, \ldots, 11\}$ eleven QE problems using VS:

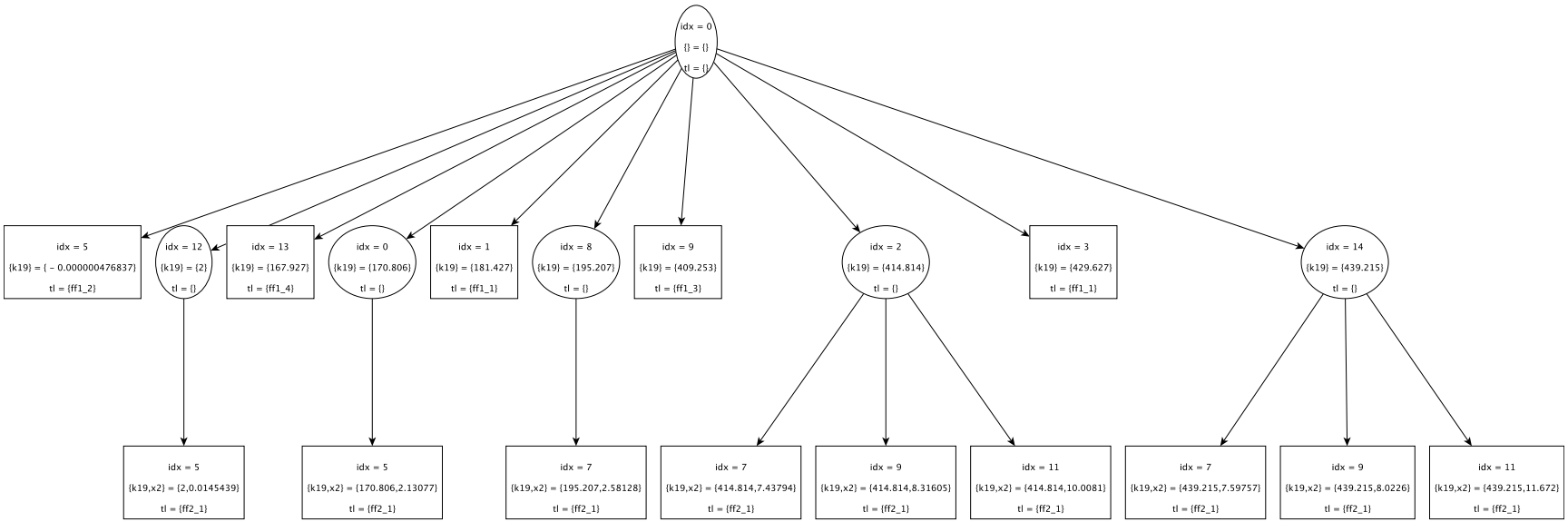
$$\varphi_{k_{19}}^{(i)} = \mathrm{VS}(\exists x_1 \ldots \exists x_{i-1} \exists x_{i+1} \ldots \exists x_{11} \varphi_{k_{19}}).$$

  Each $\varphi_{k_{19}}^{(i)}$ is a bivariate quantifier-free formula in k_{19} and the corresponding x_i.

- We then construct eleven 2-dimensional CADs, one for each $\varphi_{k_{19}}^{(i)}$ (projecting x_i and decomposing k_{19} axis).

Feasible in Redlog providing we do not extend over 0-dim k_{19}-cells. Hence accept finitely many known blind spots (a single value, hence biologically infeasible) in parameter space.

Pruned CAD tree for $\varphi^{(2)}_{k_{19}}$



- First layer decomposes k_{19}-axis.
- Rectangular cells are sections - those in top layer are the blind spots in k_{19}.
- Ovals are sectors - full dimensional cells. Over these we extend to a cylinders in the (x_2, k_{19})-plane.
- We see that the decomposition of that cylinder either has one or three sections depending on k_{19} value.

Conclusions from Approach 1

All 11 CAD trees were similar giving the following observations:

Conclusions from Approach 1

All 11 CAD trees were similar giving the following observations:

- For all positive choices of k_{19} (extending to ∞) there is at least one positive solution for $(x_1, \ldots, x_{11})$.

Conclusions from Approach 1

All 11 CAD trees were similar giving the following observations:

(i) For all positive choices of k_{19} (extending to ∞) there is at least one positive solution for $(x_1, \ldots, x_{11})$.

(ii) There is a break point around $k_{19} = 409.253$ where the system changes its qualitative behaviour:

- Below this there is exactly one solution
- Above there are at least 3 (and at most 3^{11}).
- The point itself is one of the blind spots.

Conclusions from Approach 1

All 11 CAD trees were similar giving the following observations:

(i) For all positive choices of k_{19} (extending to ∞) there is at least one positive solution for $(x_1, \ldots, x_{11})$.

(ii) There is a break point around $k_{19} = 409.253$ where the system changes its qualitative behaviour:
- Below this there is exactly one solution
- Above there are at least 3 (and at most 3^{11}).
- The point itself is one of the blind spots.

(iii) We may give the break point exactly as an algebraic number with degree 10 defining polynomial.

Approach 2: CAD + Real Triangularization

A Real Triangularization is a decomposition of a polynomial system into finitely many regular semi-algebraic systems. These are the real counterparts of the well studied *regular chains*. Such decompositions are always possible.

Approach 2: CAD + Real Triangularization

A Real Triangularization is a decomposition of a polynomial system into finitely many regular semi-algebraic systems. These are the real counterparts of the well studied *regular chains*. Such decompositions are always possible.

Consider the generic equation of degree two.

```
> R := PolynomialRing([x, c, b, a]); sys := [a x^2 + b x + c = 0]
```

$$R := polynomial\_ring$$

$$sys := [a x^2 + b x + c = 0]$$

Compute a triangular decomposition of the 4-variable hypersurface it defines.

```
> dec := RealTriangularize(sys, R) : Display(dec, R);
```

$$\left[\left\{ \begin{array}{c} a x^2 + b x + c = 0 \\ -4 a c + b^2 > 0 \text{ and } a \neq 0 \end{array} \right\}, \left\{ \begin{array}{c} 2 a x + b = 0 \\ 4 a c - b^2 = 0 \\ a \neq 0 \end{array} \right\}, \left\{ \begin{array}{c} b x + c = 0 \\ a = 0 \\ b \neq 0 \end{array} \right\}, \left\{ \begin{array}{c} c = 0 \\ b = 0 \\ a = 0 \end{array} \right\} \right]$$

Approach 2: CAD + LRT

We can also produce a Lazy Real Triangularization (LRT) which outputs the highest dimension component and unevaluated function calls: if evaluated and their output appended we gain the full solution.

Approach 2: CAD + LRT

We can also produce a Lazy Real Triangularization (LRT) which outputs the highest dimension component and unevaluated function calls: if evaluated and their output appended we gain the full solution.

```
> dec := RealTriangularize(sys, R) : Display(dec, R);
```

$$\left[\left\{\begin{array}{c} a x^2+b x+c=0 \\ -4 a c+b^2>0 \text{ and } a \neq 0 \end{array}\right., \left\{\begin{array}{c} 2 a x+b=0 \\ 4 a c-b^2=0 \\ a \neq 0 \end{array}\right., \left\{\begin{array}{c} b x+c=0 \\ a=0 \\ b \neq 0 \end{array}\right., \left\{\begin{array}{c} c=0 \\ b=0 \\ a=0 \end{array}\right.\right]$$

```
> LazyRealTriangularize(sys, R, output = piecewise)
```

$$\begin{cases} [[a x^2+b x+c=0]] & 0<-4 c a+b^2 \text{ And } a \neq 0 \\ \%LazyRealTriangularize([a=0, a x^2+b x+c=0], polynomial\_ring) & a=0 \\ \%LazyRealTriangularize([-4 a c+b^2=0, a x^2+b x+c=0], polynomial\_ring) & -4 a c+b^2=0 \\ [\,] & otherwise \end{cases}$$

LRT for Case Study with 1 free parameter

With one free parameter we can easily build an LRT for the system:

The evaluated solution component is not only triangular but:

1. With all but one equation linear in its main variable;
2. The remaining equation bivariate (one variable and the parameter);
3. Only two positivity constraints still explicitly stated (on the two variables in that bivariate equation).

Thus solving the bivariate problem allows for easy back substitution of solutions.

LRT for Case Study with 1 free parameter

With one free parameter we can easily build an LRT for the system:

The evaluated solution component is not only triangular but:

1. With all but one equation linear in its main variable;
2. The remaining equation bivariate (one variable and the parameter);
3. Only two positivity constraints still explicitly stated (on the two variables in that bivariate equation).

Thus solving the bivariate problem allows for easy back substitution of solutions.

The unevaluated components from LRT concern only a handful of isolated positive real points - so as with Approach 1 we have a few known blind spots.

Solution Formulae

With $k_{17} = 100$ and $k_{18} = 50$ the following are valid formulae for positive real solutions at all but 3 isolated points:

$$\begin{aligned}
x_{11} &= -\tfrac{1}{60}x_2^2 + \tfrac{1}{600}(10k_{19} - 10x_1 - 37x_3 + 10x_4 - 2100)x_2 \\
&\quad - \tfrac{9}{200}x_3^2 + \tfrac{1}{600}(-27x_1 + 27x_4 + 27k_{19} - 4650)x_3 \\
&\quad - x_1 + x_4 + k_{19} - 50, \\
x_{10} &= \tfrac{1}{150}x_2(x_2 + x_3 - x_4 - k_{19} + x_1 + 150), \\
x_9 &= \tfrac{1}{18200}(69x_3 + 182x_2)(x_2 + x_3 - x_4 - k_{19} + x_1 + 150), \\
x_8 &= \tfrac{15}{364}(x_2 + x_3 - x_4 - k_{19} + x_1 + 150)x_3, \\
&\vdots \\
x_2 &= \text{rational function in } x_1 \text{ and } k_{19},
\end{aligned}$$

where x_1 and k_{19} are the real positive solutions of a degree 6 bivariate polynomial equation.

Approach 2 conclusions

To finish the Approach 2 solution we can produce a full CAD sign-invariant for that bivariate polynomial. A CAD of the (x_1, k_{19})-plane into 135 cells takes a few seconds.

Approach 2 conclusions

To finish the Approach 2 solution we can produce a full CAD sign-invariant for that bivariate polynomial. A CAD of the (x_1, k_{19})-plane into 135 cells takes a few seconds.

Interrogating the cells we find the same break point value of k_{19} below which there is a single positive real solution, and above which there are exactly three positive real solutions. Again, the point itself was one the blind spots.

Approach 2 conclusions

To finish the Approach 2 solution we can produce a full CAD sign-invariant for that bivariate polynomial. A CAD of the (x_1, k_{19})-plane into 135 cells takes a few seconds.

Interrogating the cells we find the same break point value of k_{19} below which there is a single positive real solution, and above which there are exactly three positive real solutions. Again, the point itself was one the blind spots.

This time we may conclude exactly 3 (instead of at least) which indicates a possible bistability region, of interest to biologists. We also have the exact solution formulae for the region.

Approach 2 Other Choices

We can repeat this process for different choices of free parameter and different choices of fixed parameter values.

Approach 2 Other Choices

We can repeat this process for different choices of free parameter and different choices of fixed parameter values.

- With k_{17} set to 95 instead of 100 we find that the break point moves to $k_{19} = 369.917$. With k_{17} set to 105 it moves to $k_{19} = 450.077$.

Approach 2 Other Choices

We can repeat this process for different choices of free parameter and different choices of fixed parameter values.

- With k_{17} set to 95 instead of 100 we find that the break point moves to $k_{19} = 369.917$. With k_{17} set to 105 it moves to $k_{19} = 450.077$.
- Allowing k_{17} to be free and fixing $k_{19} = 200$ we find that there is only ever one positive real solution.

Approach 2 Other Choices

We can repeat this process for different choices of free parameter and different choices of fixed parameter values.

- With k_{17} set to 95 instead of 100 we find that the break point moves to $k_{19} = 369.917$. With k_{17} set to 105 it moves to $k_{19} = 450.077$.
- Allowing k_{17} to be free and fixing $k_{19} = 200$ we find that there is only ever one positive real solution.
- Allowing k_{17} to be free and fixing $k_{19} = 500$ we find the number of positive real solutions moving from 1 to 3 to 1 breaking at $k_{17} = 85.988$ and $k_{17} = 110.869$.

Approach 2 Other Choices

We can repeat this process for different choices of free parameter and different choices of fixed parameter values.

- With k_{17} set to 95 instead of 100 we find that the break point moves to $k_{19} = 369.917$. With k_{17} set to 105 it moves to $k_{19} = 450.077$.
- Allowing k_{17} to be free and fixing $k_{19} = 200$ we find that there is only ever one positive real solution.
- Allowing k_{17} to be free and fixing $k_{19} = 500$ we find the number of positive real solutions moving from 1 to 3 to 1 breaking at $k_{17} = 85.988$ and $k_{17} = 110.869$.
- Similarly, allowing k_{18} to be free and fixing $k_{19} = 200$ we find there is only ever one positive real solution; but fixing $k_{19} = 500$ instead we find 3 real solutions between $k_{18} = 44.434$ and 58.329 and 1 otherwise.

Grid Sampling

We can use grid-sampling to get an understanding of the parameter region in more than one dimension. We have considered two approaches:

Grid Sampling

We can use grid-sampling to get an understanding of the parameter region in more than one dimension. We have considered two approaches:

1. **Numeric:** Using the homotopy solver Bertini [BHSW13].

In [BDE+17] we used this to hypothesise the shape of the bistability region. However, at some sample points the method gave errors (identifying the wrong number of solutions due to rounding errors).

Grid Sampling

We can use grid-sampling to get an understanding of the parameter region in more than one dimension. We have considered two approaches:

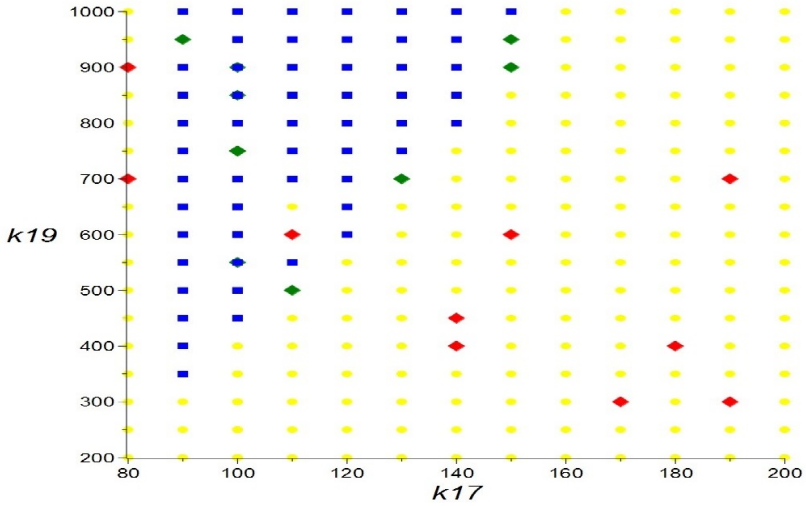
1. **Numeric:** Using the homotopy solver Bertini [BHSW13].

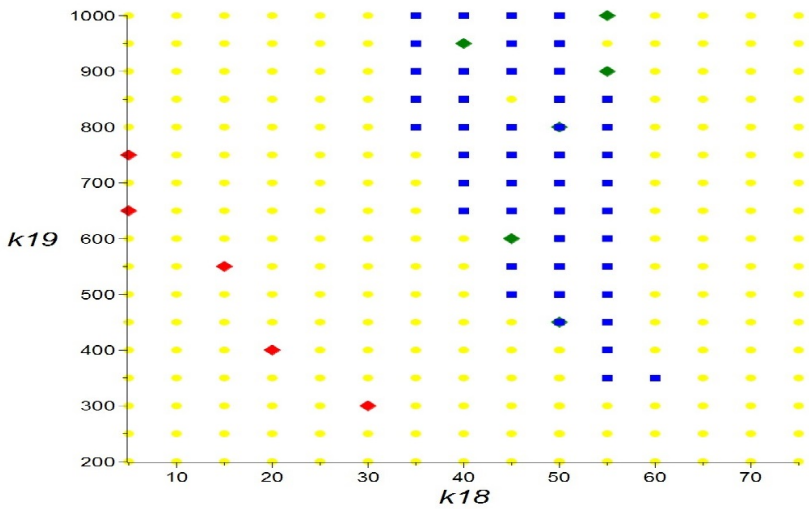
In [BDE+17] we used this to hypothesise the shape of the bistability region. However, at some sample points the method gave errors (identifying the wrong number of solutions due to rounding errors).

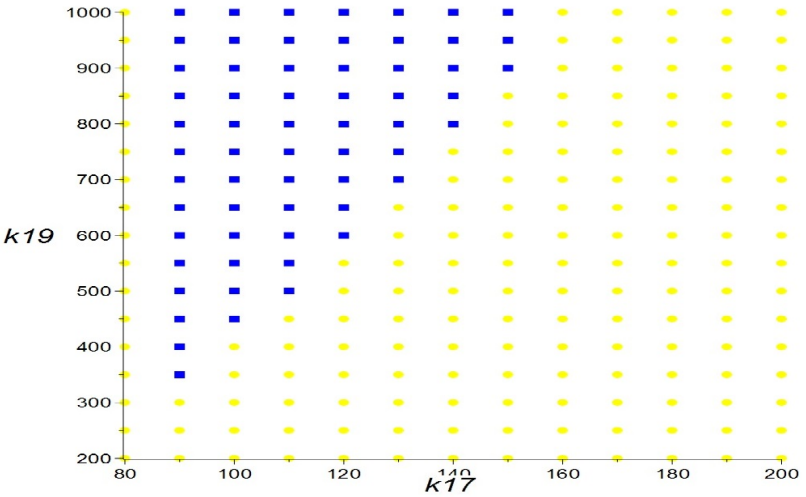
2. **Symbolic:** Iteratively applying RT + CAD with no free parameters.

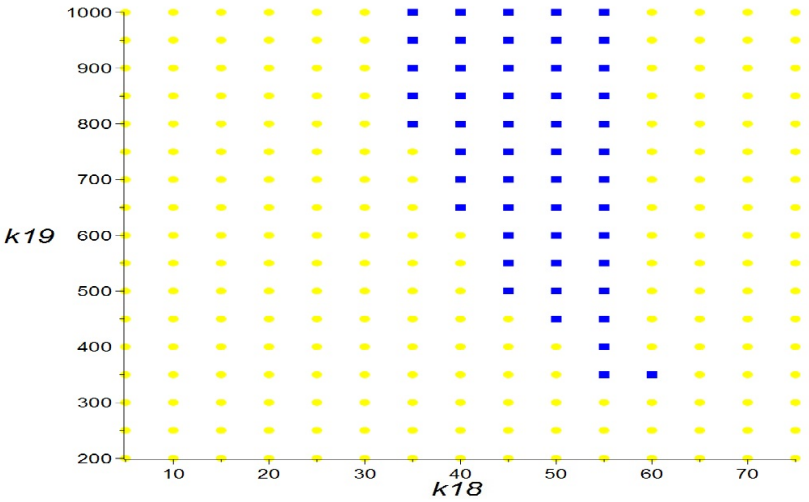
Not only did this approach avoid such errors, it even produced the images quicker than Bertini for model 26 (although the timings were reversed for Model 28). Details are in [EEG+17].

Grid Sampling Comparison



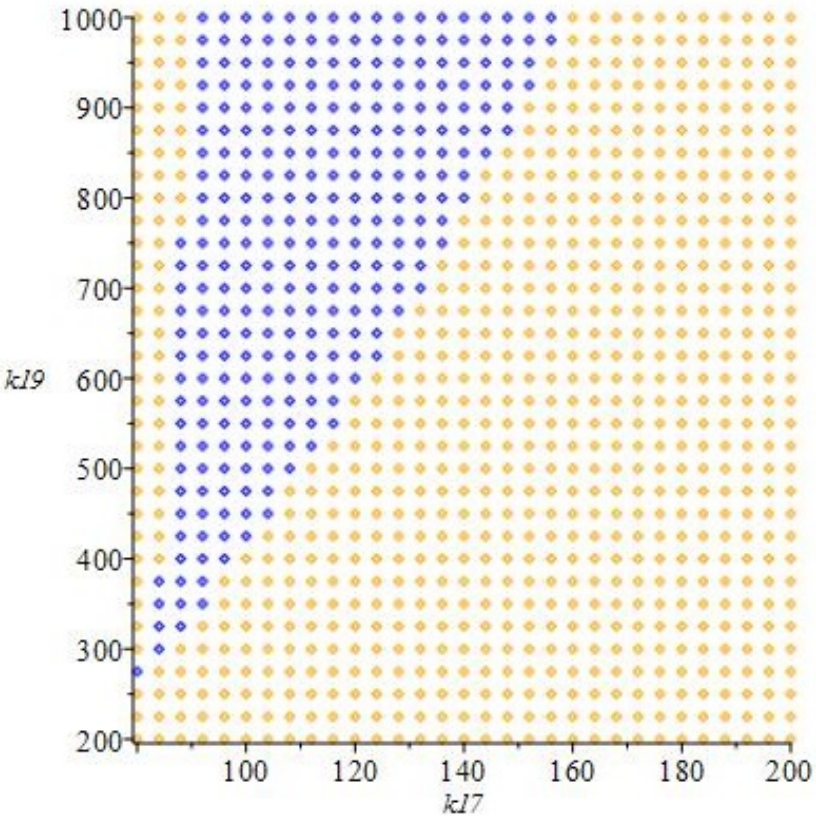


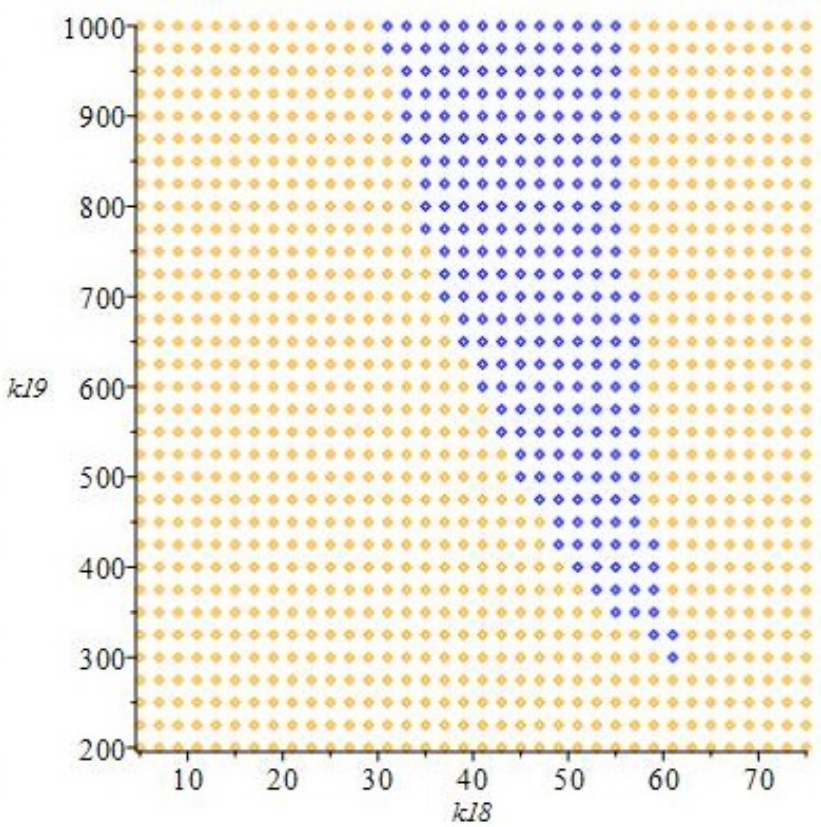




Going Further

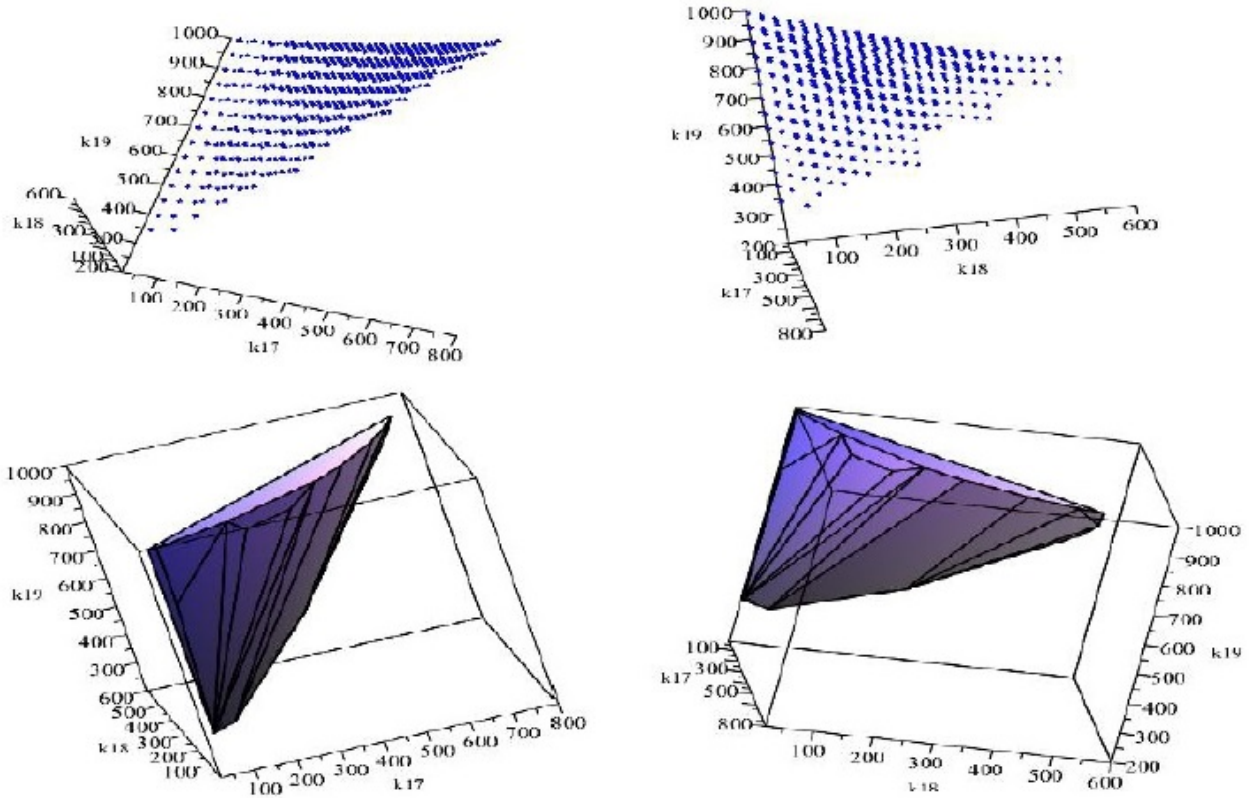
- We can increase sampling density to get a better understanding of the multi-stationarity region;





Going Further

- We can increase sampling density to get a better understanding of the multi-stationarity region;
- and make a 3d grid-sampling.



Convex Hull of the bistable points

Going Further

- We can increase sampling density to get a better understanding of the multi-stationarity region;
- and make a 3d grid-sampling.
- But ideally we want semi-algebraic descriptions. We have results [BDE+19] for two free parameters:
  - Preprocessing with a graph theoretic reduction method;
  - Lazy Real Triangularize;
  - and the restricted CAD lifting of Approach 1.

  Note: The blind spots are now blind line segments here.

Final Thoughts from Bath/Coventry/...

Conclusions:

Final Thoughts from Bath/Coventry/...

Conclusions:

- Problems like MAPK were, until recently, out of the scope of symbolic methods. But by combining the latest approaches progress is possible.

Final Thoughts from Bath/Coventry/...

Conclusions:

- Problems like MAPK were, until recently, out of the scope of symbolic methods. But by combining the latest approaches progress is possible.
- The two parameter case seems in reach: see [BDE+19]. Three parameters?

Final Thoughts from Bath/Coventry/...

Conclusions:

- Problems like MAPK were, until recently, out of the scope of symbolic methods. But by combining the latest approaches progress is possible.
- The two parameter case seems in reach: see [BDE+19]. Three parameters?
- In either case, incorporating symbolic techniques leads to much better grid sampling.

Setting for [FS20]: Hybrid Histidine-Kinase Network (HK)

$$X_1 \xrightarrow{k_1} X_2 \xrightarrow{k_2} X_3 \xrightarrow{k_3} X_4$$
$$X_3 + X_5 \xrightarrow{k_4} X_1 + X_6$$
$$X_4 + X_5 \xrightarrow{k_5} X_2 + X_6$$
$$X_6 \xrightarrow{k_6} X_5$$

$$\begin{cases} k_4x_3x_5 - k_1x_1 = 0 \\ k_5x_4x_5 + k_1x_1 - k_2x_2 = 0 \\ -k_4x_3x_5 + k_2x_2 - k_3x_3 = 0 \\ -k_4x_3x_5 - k_5x_4x_5 + k_6x_6 = 0 \\ x_1 + x_2 + x_3 + x_4 - T_1 = 0 \\ x_5 + x_6 - T_2 = 0 \end{cases}$$

- Variables x_is (concentrations of species).
- Parameters k_is (reaction rate constants) and T_i's (constants of conservation laws).

The network is called multistationary if there exists a choice of parameters for which the new system of equations has more than one positive solution.

Region of multistationarity

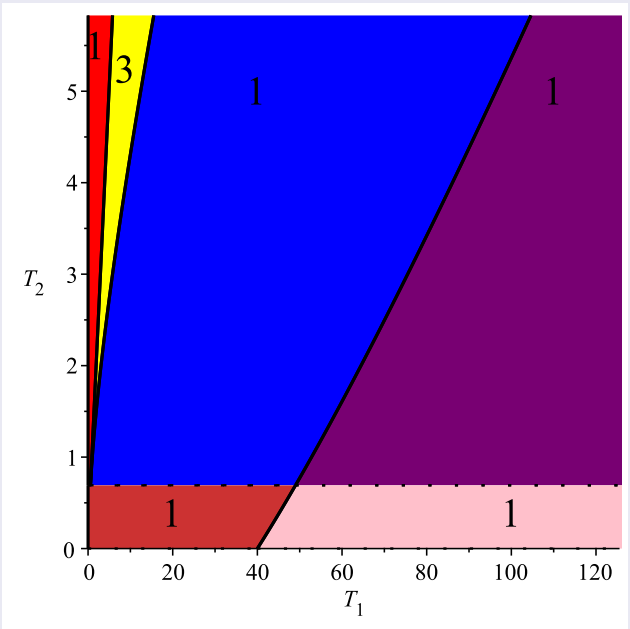
For illustration purposes fix the following values for all parameters other than T_1 and T_2.

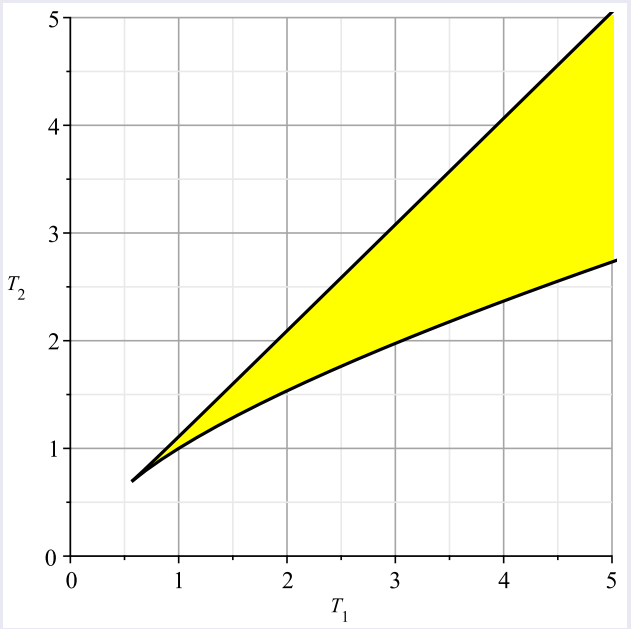
$$(k_1, \ldots, k_6) = (0.7329, 100, 73.29, 50, 100, 5).$$

Question

Find the region in (T_1, T_2)-space intersected with the box $[0, 5] \times [0, 5]$ where the network is multistationary.

Using CAD





CAD gives 6 open cells where number of steady states is invariant in each.

Number of cells grows fast, specially doubly exponential on

$$d = \text{ number of variables } + \text{ number of parameters.}$$

Therefore only applicable on very small systems.

What is Kac-Rice formula? [Kac43]

Let $f : \mathbb{R}^N \longrightarrow \mathbb{R}^N$ be a polynomial system with coefficients being polynomials on random parameters with uniform or normal distribution. Then under some conditions we can find the expected number of positive real roots:

$$\mathbb{E}\Big(\#\big(f^{-1}(0) \cap \mathbf{R}^N_{>0}\big)\Big) = \int_{\mathbf{R}^N_{>0}} \mathbb{E}\big(|\det(J_t f)| \mid f(t) = 0\big) p_t(0) dt.$$

What is Kac-Rice formula? [Kac43]

Let $f : \mathbb{R}^N \longrightarrow \mathbb{R}^N$ be a polynomial system with coefficients being polynomials on random parameters with uniform or normal distribution. Then under some conditions we can find the expected number of positive real roots:

$$\mathbb{E}\Big(\#\big(f^{-1}(0) \cap \mathbf{R}^N_{>0}\big)\Big) = \int_{\mathbf{R}^N_{>0}} \mathbb{E}\big(|\det(J_t f)| \mid f(t) = 0\big) p_t(0) dt.$$

The key to compute Kac-Rice integral in reaction network settings

For each polynomial isolate one parameters in a linear form. The easiest choice;

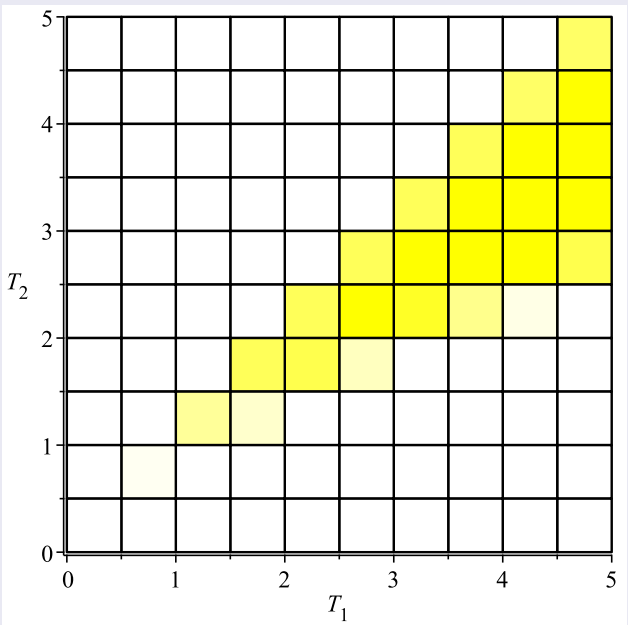
- For conservation laws isolate its conserved amount T_i.
- For steady state polynomials, choose a reaction rate constant k_i. By linear operations remove its corresponding term in the rest of steady state polynomials. Then isolate it in the only steady state polynomial containing it.

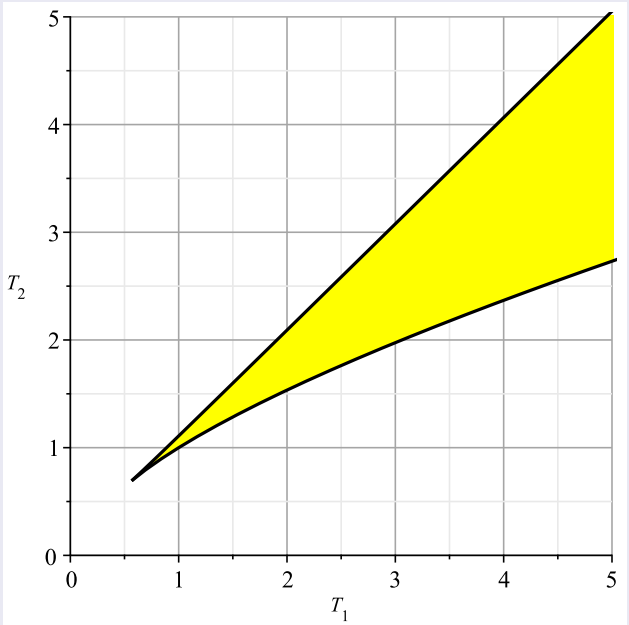
Using Kac-Rice formula

Make a grid and for each sub-box compute the Kac-Rice integral with

$$T_1 \sim U([a_i, a_{i+1}]), T_2 \sim U([b_j, b_{j+1}])$$

Using Kac-Rice formula





Make a grid and for each sub-box compute the Kac-Rice integral with

$$T_1 \sim U([a_i, a_{i+1}]),\ T_2 \sim U([b_j, b_{j+1}])$$

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.
3. Nevertheless, we can make progress, using

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.
3. Nevertheless, we can make progress, using

(a) a judicious combination of numeric and symbolic techniques

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.
3. Nevertheless, we can make progress, using

(a) a judicious combination of numeric and symbolic techniques

(*) (quite often more than one numeric and one symbolic)

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.
3. Nevertheless, we can make progress, using

(a) a judicious combination of numeric and symbolic techniques

(*) (quite often more than one numeric and one symbolic)

(b) the intelligence of my collaborators.

JHD's Personal Conclusions

1. Biology, especially the enzyme kinetics area, is a very challenging area in view of the number of parameters, and the doubly-exponential nature of symbolic algorithms (hence the DEWCAD project).
2. However, we can afford to ignore special points, as we want realistic answers.
3. Nevertheless, we can make progress, using

(a) a judicious combination of numeric and symbolic techniques

(*) (quite often more than one numeric and one symbolic)

(b) the intelligence of my collaborators.

4. The computational mathematicians are seeing slightly more acceptance by the biologists — coauthors of [BDE+17, BDE+19, EEG+17].

Bibliography


P. Aubry, D. Lazard, and M. Moreno Maza.
On the Theories of Triangular Sets.
J. Symbolic Comp., 28:105–124, 1999.


M. Bayram.
Computer Algebra Approaches to Enzyme Kinetics.
PhD thesis, University of Bath, 1993.


C.W. Brown and J.H. Davenport.
The Complexity of Quantifier Elimination and Cylindrical Algebraic Decomposition.
In C.W. Brown, editor, *Proceedings ISSAC 2007*, pages 54–60, 2007.

Bibliography

II

J.P. Bennett, J.H. Davenport, M.C. Dewar, D.L. Fisher, M. Grinfeld, and H.M. Sauro.
Computer algebra approaches to enzyme kinetics.
In Gérard Jacob and Françoise Lamnabhi-Lagarrigue, editors, *Algebraic Computing in Control*, volume 165 of *Lecture Notes in Control and Information Sciences*, pages 23–30. Springer Berlin Heidelberg, 1991.

R.J. Bradford, J.H. Davenport, M. England, H. Errami, V. Gerdt, D. Grigoriev, C. Hoyt, M. Košta, O. Radulescu, T. Sturm, and A. Weber.
A Case Study on the Parametric Occurrence of Multiple Steady States.
In *Proceedings of ISSAC '17*, pages 45–52, 2017.

Bibliography

III


R.J. Bradford, J.H. Davenport, M. England, H. Errami, V. Gerdt, D. Grigoriev, C. Hoyt, M. Košta, O. Radulescu, T. Sturm, and A. Weber.
Identifying the Parametric Occurrence of Multiple Steady States for some Biological Networks.
Journal Symbolic Comp., 98:84–119, 2019.


J.P. Bennett, J.H. Davenport, and H.M. Sauro.
Solution of Some Equations in Biochemistry.
Technical Report 88-12, University of Bath, 1988.


D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler.
Numerically Solving Polynomial Systems with Bertini.
SIAM Press, 2013.

Bibliography


B. Buchberger.
Ein Algorithmus zum Auffinden des Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal [An Algorithm for Finding a Basis for the Residue Class Ring of a Zero-Dimensional Polynomial Ideal, Tr. (M.P. Abramson) J. Symbolic Comp. **41***(2006) pp. 475–511]*.
PhD thesis, Math. Inst. University of Innsbruck, 1965.


C. Chen, J.H. Davenport, J.P. May, M. Moreno Maza, B. Xia, and R. Xiao.
Triangular decomposition of semi-algebraic systems.
J. Symbolic Comp. Available online 22 December 2011, 2011.

Bibliography

V


D.A. Cox, J. Little, and D. O'Shea.
Ideals, Varieties, and Algorithms (Fourth Edition).
Undergraduate Texts in Mathematics. Springer, Heidelberg, 2015.


C. Chen and M. Moreno Maza.
Quantifier Elimination by Cylindrical Algebraic Decomposition Based on Regular Chains.
In K. Nabeshima, editor, *Proceedings ISSAC 2014*, pages 91–98, 2014.


C. Chen and M. Moreno Maza.
Quantifier elimination by cylindrical algebraic decomposition based on regular chains.
J. Symbolic Comp., 75:74–93, 2016.

Bibliography

VI


G.E. Collins.
Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition.
In *Proceedings 2nd. GI Conference Automata Theory & Formal Languages*, pages 134–183, 1975.


J.H. Davenport and J. Heintz.
Real Quantifier Elimination is Doubly Exponential.
J. Symbolic Comp., 5:29–35, 1988.


A. Dolzmann, A. Seidl, and Th. Sturm.
Redlog User Manual, edition 3.0.
`http://www.fmi.unipassau.de/~seidl/publications/DSS04b`, 2004.

Bibliography

VII


M. England, H. Errami, D. Grigoriev, O. Radulescu, T. Sturm, and A. Weber.
Symbolic Versus Numerical Computation and Visualization of Parameter Regions for Multistationarity of Biological Networks.
In *Proceedings CASC 2017*, pages 93–108, 2017.


E. Feliu and AH. Sadeghimanesh.
Kac-Rice formulas and the number of solutions of parametrized systems of polynomial equations.
`https://arxiv.org/abs/2010.00804`, 2020.


M. Kac.
On the Average Number of Real Roots of a Random Algebraic Equation.
Bull. A.M.S., 49:314–320, 1943.

Bibliography

VIII


E. Mayr and A. Meyer.
The Complexity of the Word Problem for Commutative Semi-groups and Polynomial Ideals.
Adv. in Math., 46:305–329, 1982.


E.W. Mayr and S. Ritscher.
Dimension-dependent bounds for Gröbner bases of polynomial ideals.
J. Symbolic Comp., 49:78–94, 2013.


Z. Tonks.
Poly-algorithmic Techniques in Real Quantifier Elimination.
PhD thesis, University of Bath, 2021.

Bibliography

IX


V. Weispfenning.
The Complexity of Linear Problems in Fields.
J. Symbolic Comp., 5:3–27, 1988.


V. Weispfenning.
Quantifier elimination for real algebra — the cubic case.
In *Proceedings ISSAC 1994*, pages 258–263, 1994.


Wu and Wen Tsün.
Mechanical Theorem Proving in elementary differential geometry.
Kexue Tongbao 9, 23:523–4, 1978.